

Solution (With great power comes great responsibility)

1)

```
def fib(x: Int): BigInt = {  
    def fibHelper(x: Int, prev: BigInt = 0, next: BigInt = 1): BigInt = x match {  
        case 0 => prev  
        case 1 => next  
        case _ => fibHelper(x - 1, next, (next + prev))  
    }  
    fibHelper(x)  
}
```

2)

```
def meanElements(list: List[Float]): Float = {  
    list match {  
        case x :: tail => (x + meanElements(tail) * list.length) / list.length  
        case Nil => 0  
    }  
}
```

3)

```
Def  
gcd(a: Int, b: Int): Int = {  
    If  
    (b == 0) a  
    Else  
    gcd(b, a % b)  
}
```

4)

```
def reverse[A](in: List[A], out: List[A]): List[A] = in match {  
    case Nil => out
```

```
case Cons(h,t) => reverse(t, append(List(h), out))
}
```

5)

```
def drop[A](l: List[A], n: Int): List[A] = l match {
  case Nil => Nil
  case _ if n == 0 => l
  case Cons(_, tail) => drop(tail, n-1)
}
```